MULTIMEDIA UNIVERSITY

STUDENT ID NO

MULTIMEDIA UNIVERSITY

FINAL EXAMINATION

TRIMESTER 3, 2016/2017

DMT5018 – MICROCONTROLLER TECHNOLOGY

(DEE)

30 MAY 2017
9.00am – 11.00am
(2 Hours)

INSTRUCTIONS TO STUDENT

1. This question paper consists of **6** pages (**3** pages with **4** questions and **3** pages for appendix)
2. Answer **ALL** questions. All necessary working steps must be shown.
3. Write all your answers in the answer booklet provided.

QUESTION 1 [25 Marks]

a) Define computer, and draw the flowchart of program execution process in a computer.

[6 marks]

b) State the differences between a microcontroller and a microprocessor in the perspective of architecture, instruction set features, price and application.

[12 marks]

c) List three products that are most likely to apply a microcontroller.

[3 marks]

d) Fill in the missing steps for a typical memory write operation below.

Step 1: CPU sends memory request control signal (for memory operation)
Step 2: _____
Step 3: _____
Step 4: CPU sends write control signal, indicating valid data in on the data bus.
Step 5: _____
Step 6: _____

[4 marks]

QUESTION 2 [25 Marks]

a) List the four control pins available in 8051 microcontroller.

[4 marks]

b) Referring to the assembly language program below, state the content of stack pointer (SP), R2, R3 and R4 after executing the program.

```
MOV  SP, #40H
MOV  R2, #56H
MOV  R3, #0F3H
MOV  R4, #1CH
PUSH 4
PUSH 2
PUSH 3
POP  2
POP  4
POP  3
```

[8 marks]

Continued...

c) Referring to the assembly language program below, answer the following questions.

```
Line 1    MOV  PSW, #08H
Line 2    MOV  A, #47H
Line 3    MOV  R1, #23H
Line 4    ADD  A, #89H
Line 5    MOV  B, #10H
Line 6    MUL  AB
Line 7    MOV  B, R1
Line 8    DIV  AB
```

(i) State the register bank in use.

[1 mark]

(ii) State the content of accumulator (ACC) and Program Status Word (PSW) after executing the instruction in Line 4.

[6 marks]

(iii) State the content of accumulator (ACC), register B and overflow flag (OV) after executing the instruction in Line 6.

[3 marks]

(iv) State the content of accumulator (ACC), register B and overflow flag (OV) after executing the instruction in Line 8.

[3 marks]

Continued...

QUESTION 3 [25 Marks]

Referring to the assembly language program below, answer the following questions.

Program Counter	Label	Instruction	Opcode	Addressing Mode	Instruction Type
		ORG 20H			
(i)		MOV A, #80H	(v)	(ix)	(xii)
(ii)	BACK:	MOV P1, A	(vi)	(x)	(xiii)
(iii)		RR A	(vii)		(xiv)
(iv)		SJMP BACK	(viii)	(xi)	(xv)
		END			

a) Give the answers for all cells marked with roman numbers (i) to (xv).

[15 marks]

b) Assuming Port 1 is connected to LEDs, and the function of the program is to create a running LED from left to right (P1.7 to P1.0) infinitely, rewrite the program so that the rotation from left to right is limited to 10 times only.

[5 marks]

c) Calculate the program execution time of the program written in Question 3(b), assuming the oscillator frequency used is 11.0592 MHz.

[5 marks]

QUESTION 4 [25 Marks]

a) Write an assembly language program, without using interrupt, to display the number 1, 2 and 3 repetitively on a 7-segment display connected at Port 1. Use Timer 0 to provide 0.5 s delay for each number display. Assume common anode 7-segment display is used, and 12 MHz oscillator frequency in operation.

[12 marks]

b) Write an assembly language program, using interrupt, to continuously transmit the character 'Z' through serial port, and to store any receive data through serial port at R1 of Bank 2. Use 4800 baud, assume SMOD = 0 and oscillator frequency of 11.0592 MHz.

[13 marks]

End of Page.

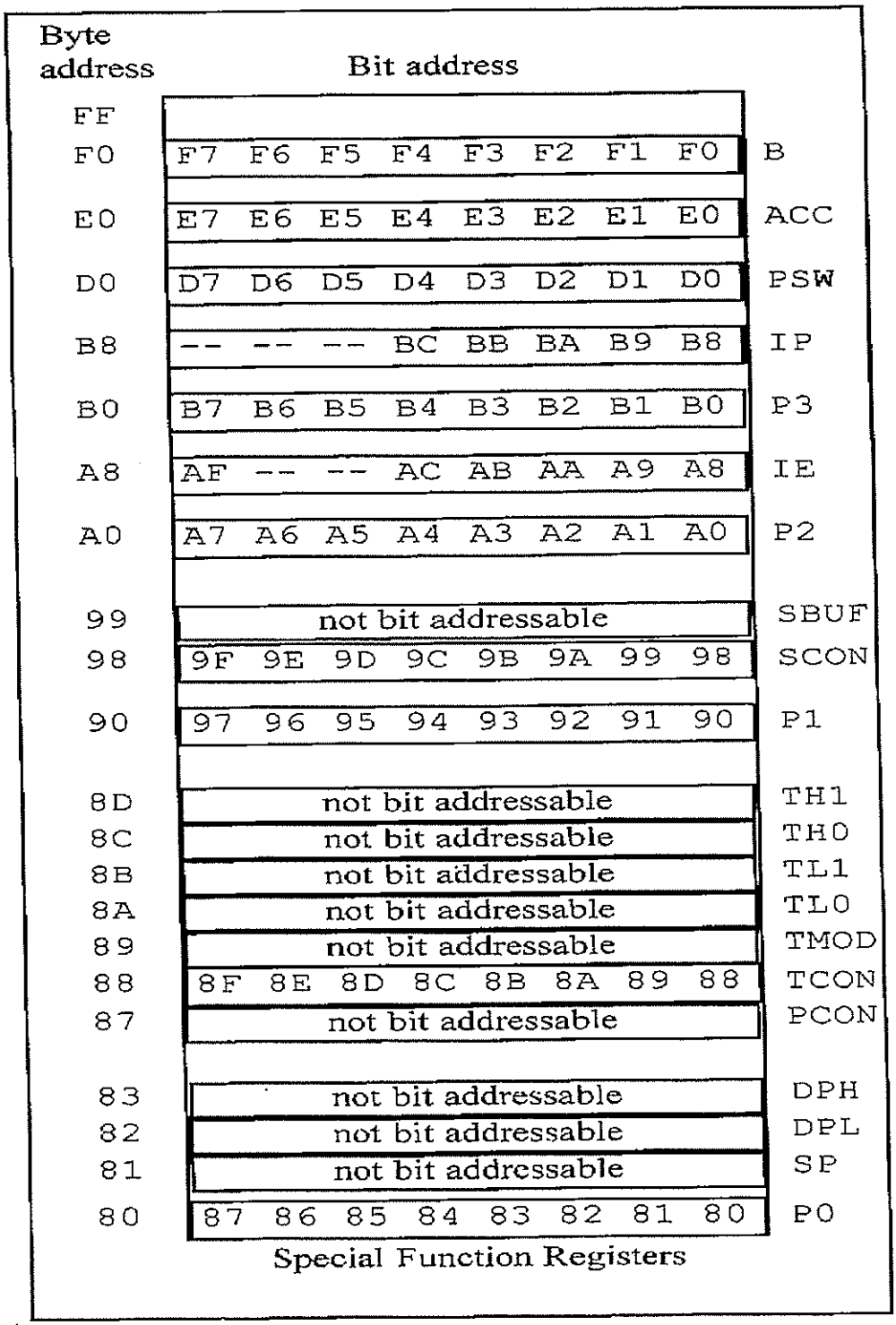
APPENDIX A: OPCODE MAP

Instruction Code Summary

L \ H	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0	NOP	JBC bit, rel	JB bit, rel	JNB bit, rel	JC rel	JNC rel	JZ rel	JNZ rel	SJMP rel	MOV DPTR, # data 16	ORL C, /bit	ANL C, /bit	PUSH dir	POP dir	MOVX A, @DPTR	MOVX @DPTR, A
1	AJMP (P0)	ACALL (P0)	AJMP (P1)	ACALL (P1)	AJMP (P2)	ACALL (P2)	AJMP (P3)	ACALL (P3)	AJMP (P4)	ACALL (P4)	AJMP (P5)	ACALL (P5)	AJMP (P6)	ACALL (P6)	AJMP (P7)	ACALL (P7)
2	LJMP addr16	LCALL addr16	RET	RETI	ORL dir, A	ANL dir, A	XRL dir, A	ORL C, bit	ANL C, bit	MOV bit, C	MOV C, bit	CPL bit	CLR bit	SETB bit	MOVX A, @R0	MOVX @R0, A
3	RR A	RRC A	RL A	RLC A	ORL dir, # data	ANL dir, # data	XRL dir, # data	JMP @A+DPTR	MOVC A, @A+PC	MOVC A, @A+DPTR	INC DPTR	CPL C	CLR C	SETB C	MOVX A, @R1	MOVX @R1, A
4	INC A	DEC A	ADD A, # data	ADDC A, # data	ORL A, # data	ANL A, # data	XRL A, # data	MOV A, # data	DIV AB	SUBB A, # data	MUL AB	CJNE A, # data, rel	SWAP A	DA A	CLR A	CPL A
5	INC dir	DEC dir	ADD A, dir	ADDC A, dir	ORL A, dir	ANL A, dir	XRL A, dir	MOV dir, # data	MOV dir, dir	SUBB A, dir		CJNE A, dir, rel	XCH A, dir	DJNZ dir, rel	MOV A, dir	MOV dir, A
6	INC @R0	DEC @R0	ADD A, @R0	ADDC A, @R0	ORL A, @R0	ANL A, @R0	XRL A, @R0	MOV @R0, # data	MOV dir, @R0	SUBB A, @R0	MOV @R0, dir	CJNE @R0, # data, rel	XCH A, @R0	XCHD A, @R0	MOV A, @R0	MOV @ R0, A
7	INC @R1	DEC @R1	ADD A, @R1	ADDC A, @R1	ORL A, @R1	ANL A, @R1	XRL A, @R1	MOV @R1, # data	MOV dir, @R1	SUBB A, @R1	MOV @R1, dir	CJNE @R1, # data, rel	XCH A, @R1	XCHD A, @R1	MOV A, @R1	MOV @ R1, A
8	INC R0	DEC R0	ADD A, R0	ADDC A, R0	ORL A, R0	ANL A, R0	XRL A, R0	MOV R0, # data	MOV dir, R0	SUBB A, R0	MOV R0, dir	CJNE R0, # data, rel	XCH A, R0	DJNZ R0, rel	MOV A, R0	MOV R0, A
9	INC R1	DEC R1	ADD A, R1	ADDC A, R1	ORL A, R1	ANL A, R1	XRL A, R1	MOV R1, # data	MOV dir, R1	SUBB A, R1	MOV R1, dir	CJNE R1, # data, rel	XCH A, R1	DJNZ R1, rel	MOV A, R1	MOV R1, A
A	INC R2	DEC R2	ADD A, R2	ADDC A, R2	ORL A, R2	ANL A, R2	XRL A, R2	MOV R2, # data	MOV dir, R2	SUBB A, R2	MOV R2, dir	CJNE R2, # data, rel	XCH A, R2	DJNZ R2, rel	MOV A, R2	MOV R2, A
B	INC R3	DEC R3	ADD A, R3	ADDC A, R3	ORL A, R3	ANL A, R3	XRL A, R3	MOV R3, # data	MOV dir, R3	SUBB A, R3	MOV R3, dir	CJNE R3, # data, rel	XCH A, R3	DJNZ R3, rel	MOV A, R3	MOV R3, A
C	INC R4	DEC R4	ADD A, R4	ADDC A, R4	ORL A, R4	ANL A, R4	XRL A, R4	MOV R4, # data	MOV dir, R4	SUBB A, R4	MOV R4, dir	CJNE R4, # data, rel	XCH A, R4	DJNZ R4, rel	MOV A, R4	MOV R4, A
D	INC R5	DEC R5	ADD A, R5	ADDC A, R5	ORL A, R5	ANL A, R5	XRL A, R5	MOV R5, # data	MOV dir, R5	SUBB A, R5	MOV R5, dir	CJNE R5, # data, rel	XCH A, R5	DJNZ R5, rel	MOV A, R5	MOV R5, A
E	INC R6	DEC R6	ADD A, R6	ADDC A, R6	ORL A, R6	ANL A, R6	XRL A, R6	MOV R6, # data	MOV dir, R6	SUBB A, R6	MOV R6, dir	CJNE R6, # data, rel	XCH A, R6	DJNZ R6, rel	MOV A, R6	MOV R6, A
F	INC R7	DEC R7	ADD A, R7	ADDC A, R7	ORL A, R7	ANL A, R7	XRL A, R7	MOV R7, # data	MOV dir, R7	SUBB A, R7	MOV R7, dir	CJNE R7, # data, rel	XCH A, R7	DJNZ R7, rel	MOV A, R7	MOV R7, A

2Byte 3Byte
2Cycle 4Cycle

APPENDIX B: 8051 SPECIAL FUNCTION REGISTER (SFR)


Byte address
Bit address
FF
F0 F7 F6 F5 F4 F3 F2 F1 F0 B
E0 E7 E6 E5 E4 E3 E2 E1 E0 ACC
D0 D7 D6 D5 D4 D3 D2 D1 D0 PSW
B8 -- -- -- BC BB BA B9 B8 IP
B0 B7 B6 B5 B4 B3 B2 B1 B0 P3
A8 AF -- -- AC AB AA A9 A8 IE
A0 A7 A6 A5 A4 A3 A2 A1 A0 P2
99 not bit addressable SBUF
98 9F 9E 9D 9C 9B 9A 99 98 SCON
90 97 96 95 94 93 92 91 90 P1
8D not bit addressable TH1
8C not bit addressable TH0
8B not bit addressable TL1
8A not bit addressable TL0
89 not bit addressable TMOD
88 8F 8E 8D 8C 8B 8A 89 88 TCON
87 not bit addressable PCON
83 not bit addressable DPH
82 not bit addressable DPL
81 not bit addressable SP
80 87 86 85 84 83 82 81 80 P0
Special Function Registers

SFR RAM Address (Byte and Bit)

APPENDIX C: BIT-ADDRESSABLE RAM LOCATIONS

Byte Address	Bit Address							
2F	7F	7E	7D	7C	7B	7A	79	78
2E	77	76	75	74	73	72	71	70
2D	6F	6E	6D	6C	6B	6A	69	68
2C	67	66	65	64	63	62	61	60
2B	5F	5E	5D	5C	5B	5A	59	58
2A	57	56	55	54	53	52	51	50
29	4F	4E	4D	4C	4B	4A	49	48
28	47	46	45	44	43	42	41	40
27	3F	3E	3D	3C	3B	3A	39	38
26	37	36	35	34	33	32	31	30
25	2F	2E	2D	2C	2B	2A	29	28
24	27	26	25	24	23	22	21	20
23	1F	1E	1D	1C	1B	1A	19	18
22	17	16	15	14	13	12	11	10
21	0F	0E	0D	0C	0B	0A	09	08
20	07	06	05	04	03	02	01	00

APPENDIX D: PROGRAM STATUS WORD (PSW)

CY	AC	F0	RS1	RS0	OV	--	P

APPENDIX E: TIMER/COUNTER MODE CONTROL (TMOD) REGISTER

G	C/T	M1	M0	G	C/T	M1	M0

APPENDIX F: SERIAL CONTROL (SCON) REGISTER

SM0	SM1	SM2	REN	TB8	RB8	TI	RI

APPENDIX G: INTERRUPT ENABLE (IE) REGISTER

EA	--	ET2	ES	ET1	EX1	ET0	EX0

APPENDIX H: INTERRUPT PRIORITY (IP) REGISTER

--	--	PT2	PS	PT1	PX1	PT0	PX0